数理解析研究所講究録
156巻 1972年 69-82

Indexed Grammar の木構造について
ー uvwxy 定理の拡張ー

京大 理　林 健志

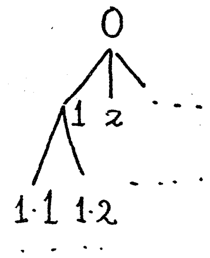
§1. 序

CF文法に対する uvwxy 定理の Stack Automaton に対する拡張は，既に Ogden [1968] によって得られている。能力的に Stack Automaton を含む Indexed Grammar に対する拡張を得た。この拡張定理を応用して，ある種の言語，例えば，$\{a^{n!} \mid n \geq 1\}$ や $\{(\#w)^{|w|} \mid w \in \{a,b\}^*\}$ が Indexed Language でないことが証明出来る。拡張定理は Indexed Grammar の生成木の増加の様子を述べた形をしているので，まず生成木を formal に扱う必要がある。そのために，CF文法の生成木を記述するために，Brainerd [1969], Takahashi [1970] らが採用した方法を基礎とした。

§2. 木構造の記述法　I.【基本的な定義】

[Def 2-1] J を自然数の集合とする。J^* を J によって生成された free monoid とし，2項演算を'・，単位元を 0 で示す。

次の(i),(ii)を満たすJ^*の有限部分集合Dのことを<u>tree domain</u>という。



[J^*の図][図1]

(i) $p \cdot q \in D$ なら $p \in D$

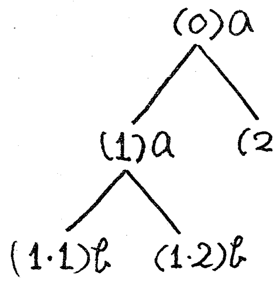
(ii) $p \cdot j \in D$, $1 \leq i \leq j$ なら $p \cdot i \in D$

Dの元を<u>頂点(node)</u>といい, pと$p \cdot i$がDの元なら$p \cdot i$をpの<u>子(direct descendant)</u>という。(i)より($D \neq \phi$ならば)$0 \in D$である。頂点0をDの<u>根(root)</u>という。

[Def 2-2] Σをアルファベット(空でない記号の集合)とした時, <u>Σ上のtree</u>とは写像$\gamma : D \to \Sigma$のことをいう。ただしDはtree domainである。$\gamma(p)$を頂点pの<u>名称(label)</u>という。γのdomain Dを$dom(\gamma)$で示す。<u>T_Σ</u>でΣ上のtree全体からなる集合を示す。以下写像γとそのgraphを同一視する。

[例1] $\Sigma = \{a, b\}$, $\gamma = \{(0,a), (1,a), (2,b), (1 \cdot 1, b), (1 \cdot 2, b)\}$

とすると, γはΣ上のtreeで, 通常γは図2のように図示される。treeの頂点の縦の関係$\lessdot$, と横の関係$\prec$を定義する。



[図2]

[Def 2-3] $p, q \in J^*$に対して,

$$p \leq q \quad \text{iff} \quad (\exists r \in J^*)(q = p \cdot r)$$

$$p \lessdot q \quad \text{iff} \quad p \leq q \wedge p \neq q$$

$$p \prec q \quad \text{iff} \quad (\exists r \in J^*)(\exists i \in J)(\exists j \in J)\begin{pmatrix} i < j, \\ r \cdot i \leq p,\ r \cdot j \leq q \end{pmatrix}$$

$p \lessdot q$ の時 q は p の子孫であるといい，$p \gtrdot q$ の時，q は p の右にあるという。

〔例2〕 $1 \lessdot 1\cdot1$，$1\cdot1 \gtrdot 1\cdot2$，$1 \not\lessdot 2\cdot1$

〔Def 2-4〕 $\gamma \in T_\Sigma$ に対して γ の先端(front) $\hat{\gamma}$ とは，

$$\hat{\gamma} \underset{\text{def}}{=} \{ (p, a) \in \gamma \mid \text{for any } q \in dom(\gamma),\ p \not\lessdot q \}$$

なる集合のことである。又根から先端の頂点へ到る道を考えたいことがある。すなわち，γ の要素の列

$$\langle (p_0, a_0), \cdots, (p_i, a_i), \cdots, (p_l, a_l) \rangle$$

が γ の鎖(chain)であるとは，

$$p_0 = 0,\quad p_i = p_{i-1}\cdot j_i,\ j_i \in J\ (i=1,\cdots,l),\ (p_l, a_l) \in \hat{\gamma}$$

の時にいう。l のことを鎖の長さと呼ぶ。

Ⅱ. 〚tree に関する種々の操作〛

名称が付いた Σ 上の tree γ に対しては，その subtree などが自然に考えられる。以下 tree に関する種々の operation を導入するが，名称は元の tree γ から自然に導びかれるものになっている。

〔Def 2-5〕 $\gamma \in T_\Sigma$ に対して，

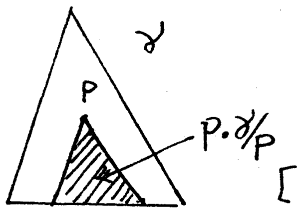
1) $\gamma/p \underset{\text{def}}{=} \{ (q, a) \mid (p\cdot q, a) \in \gamma \}$ $(p \in dom(\gamma))$

2) $p\cdot\gamma \underset{\text{def}}{=} \{ (p\cdot q, a) \mid (q, a) \in \gamma \}$ $(p \in J^*)$

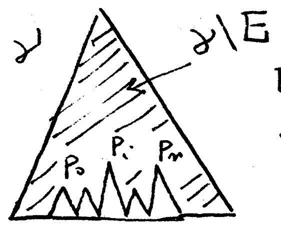
3) $E \subset dom(\gamma)$ に対して，

$$\gamma \backslash E \underset{\text{def}}{=} \gamma - \bigcup_{p \in E} p\cdot\gamma/p$$

常に γ/p, $\gamma\backslash E \in \mathcal{T}_\Sigma$ である。[図3],[図4]参照



[図3]



$E=\{p_0,\cdots,p_n\}$ s.t. $p_i \ni p_{i+1}$ $(0 \leq i \leq n-1)$ の時

[図4]

[Def 2-6] <u>木による置換</u>とは次の操作を言う。

$\gamma, \tau_0, \cdots, \tau_n \in \mathcal{T}_\Sigma$, $p_i \in dom(\gamma)$ $(i=0,\cdots,n)$ に対して, $p_i \ni p_{i+1}$ $(i=0,\cdots,n-1)$ かつ $\tau_i(0)=\gamma(p_i)$ $(i=0,\cdots,n)$ が成立している時, $\mathcal{T}_\Sigma$ の元

$$\gamma\begin{bmatrix} p_0, p_1, \cdots, p_n \\ \tau_0, \tau_1, \cdots, \tau_n \end{bmatrix} \underset{def}{=\!=} \gamma\backslash E \cup \bigcup_{i=0}^{n} p_i \cdot \tau_i$$

但し, $E=\{p_0,\cdots,p_n\}$

を木による置換によって得る。

[Def 2-7] tree の先端の名称を左から右に concatenation をして出来る string が必要になってくる。これを与える <u>yield function</u> $g: \mathcal{T}_\Sigma \rightarrow \Sigma^*$ を帰納的に定義する。

i) $g(\gamma)=\gamma(0)$ if $dom(\gamma)=\{0\}$

ii) $g(\gamma)=g(\gamma/1)\cdots g(\gamma/j)$

if $1,\cdots,j \in dom(\gamma)$, $j+1 \notin dom(\gamma)$

§3 Indexed Grammar の木構造

Indexed Grammar の基本的な定義, 及び一般的な表示法に関しては, Aho [1968] に従った。

[Def 3-1] <u>Indexed Grammar</u>とは次のような5組，

$G=(N, T, F, P, S)$のことである。

i) Nは空でない有限集合で，<u>非終端アルファベット</u>と呼ばれ，その元を<u>非終端記号(nonterminal)</u>という。

ii) Tは空でない有限集合で，$N \cap T = \phi$を満たし，<u>終端アルファベット</u>と呼ばれ，その元を<u>終端記号(terminal)</u>という。

iii) Fは有限集合でその各要素は$N \times (N \cup T)^*$の有限部分集合である。$f \in F$を<u>index</u>といい，$(A, \chi) \in f$を$A \to \chi$と書いて<u>index production in f</u>という。

iv) Pは$N \times (NF^* \cup T)^*$の有限部分集合である。

$(A, \alpha) \in P$を$A \to \alpha$と書いて<u>production</u>と呼ぶ。

v) SはNの1つの元で<u>sentence symbol</u>という。

以下，$N \cup T$をVと書き，$V_F = NF^* \cup T \cup \{\varepsilon\}$とした時，$\mathcal{T}_{V_F}$で考える。(treeの名称の集合として$V_F$をとるわけである。)

[Def 3-2] Indexed Grammar (以下 IGと略す。)

$G=(N, T, F, P, S)$が与えられた時，$\mathcal{T}_{V_F}$上のrelation $\vdash_G$を次のように定義する。$\gamma, \delta \in \mathcal{T}_{V_F}$に対して，$\gamma \vdash_G \delta$

iff 1. (index expanding)

$\exists (p, A\xi) \in \hat{\gamma}$，$\exists A \longrightarrow X_1 \eta_1 X_2 \eta_2 \cdots X_k \eta_k \in P$で

$\delta = \gamma \cup \bigcup_{j=1}^{k} \{(p \cdot j, X_j \mu_j)\}$の時。ここで$\mu_j = \begin{cases} \eta_j \xi & \text{if } X_j \in N \\ \varepsilon & \text{if } X_j \in T \end{cases}$

ただし　$k=0$の時 (i.e $A \to \varepsilon$)は$\delta = \gamma \cup \{(p \cdot 1, \varepsilon)\}$

又は, 2. (index consuming)

$\exists (P, Af\xi) \in \hat{\gamma}$, $\exists A \to X_1 X_2 \cdots X_k \in f \in F$ で

$\delta = \gamma \cup \bigcup_{j=1}^{k} \{(P\cdot j, X_j \mu_j)\}$ の時, ここで $\mu_j = \begin{cases} \xi & \text{if } X_j \in N \\ \varepsilon & \text{if } X_j \in T \end{cases}$

ただし $k = 0$ の時 (i.e $A \to \varepsilon \in f$) は $\delta = \gamma \cup \{(P\cdot 1, \varepsilon)\}$

$\overset{*}{\underset{G}{\vdash}}$ で $\underset{G}{\vdash}$ の $\mathcal{T}_{V_F}$ 上の reflexive and transitive closure を示す。(注)

以上の定義, 及びこれからは特にことわらない限り次の symbolic convention に従っている.

(1) A, B, は N の元。(2) f は F の元。(3) η, ξ, μ は F^* の元。

(4) X は V の元。(5) ε は空語 (6) P, q, r は J^* の元。

$$\mathcal{T}(G) \underset{\text{def}}{=} \{\gamma \in \mathcal{T}_{V_F} \mid \{(0, S)\} \overset{*}{\underset{G}{\vdash}} \gamma,\ g(\gamma) \in T^*\}$$ を

G の生成木 (derivation tree) の集合という。又,

$L(G) \underset{\text{def}}{=} g(\mathcal{T}(G))$ を G で生成 (generate) された言語

という。(この定義は Aho による $L(G)$ の定義と一致している.)

$L \subset T^*$ に対して $L = L(G)$ なる IG : G が存在する時,

L を indexed language と呼ぶ。

IG : G の木構造とは,

$$\tilde{\mathcal{T}}(G) \underset{\text{def}}{=} \{\gamma \in \mathcal{T}_{V_F} \mid \gamma = \Lambda, \text{又は } \{(0, A\eta)\} \overset{*}{\underset{G}{\vdash}} \gamma, \exists A \in N, \exists \eta \in F^*\}$$

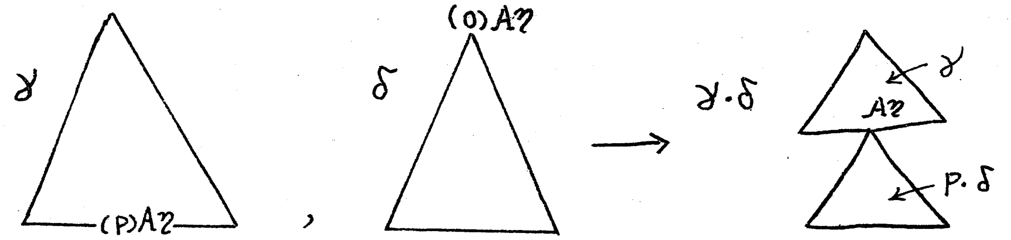
のことである。ここで Λ は empty tree $\Lambda : \phi \to V_F$ のことである。

明らかに $\mathcal{T}(G) \subset \tilde{\mathcal{T}}(G)$ である。

(注) すなわち $\gamma \overset{*}{\underset{G}{\vdash}} \delta \iff \gamma = \delta$ or $\exists \gamma_0, \ldots, \gamma_n$ s.t $\gamma_0 = \gamma$, $\gamma_n = \delta$
$\gamma_i \underset{G}{\vdash} \gamma_{i+1}$ $(0 \le i \le n-1)$

[Def 3-3] $\gamma, \delta \in \tilde{\mathcal{T}}(G)$ で, $g(\gamma) \in T^* A \eta T^*$ かつ, $\delta = \Lambda$, 又は $\delta(0) = A\eta$ の時, $\gamma \cdot \delta \in \tilde{\mathcal{T}}(G)$ を定義。

$$\gamma \cdot \delta = \begin{cases} \gamma & \text{if} \quad \delta = \Lambda \\ \gamma\left[\begin{smallmatrix} p \\ \delta \end{smallmatrix}\right] & \text{if} \quad \delta(0) = A\eta, \text{ ここで } (p, A\eta) \in \hat{\gamma} \end{cases}$$



[図5]

§4. 生成木増加定理

今まで, 述べて来た tree の記述法を使えば, 定理の証明, 及び記述に必要な, 色々の概念, 関数を formal に定義できる。証明は非常に長いのでここでは与えないが, 基本的な idea を説明するために必要な定義を2, 3あげておく。詳細に関しては筆者の論文(林[1972])を参照されたい。

[Def 4-1] $\gamma \in \tilde{\mathcal{T}}(G)$, $p \in \mathrm{dom}(\gamma)$ に対して,
$\pi_{1\gamma} : \mathrm{dom}(\gamma) \to V \cup \{\varepsilon\}$, $\pi_{2\gamma} : \mathrm{dom}(\gamma) \to F^*$, $\tilde{\pi}_{2\gamma} : \mathrm{dom}(\gamma) \to F \cup \{\varepsilon\}$
を次のように定義する。

$$\left.\begin{array}{l} \pi_{1\gamma}(p) = X \\ \pi_{2\gamma}(p) = \eta \end{array}\right\} \quad \text{if} \quad \gamma(p) = X\eta$$

(ただし $\gamma(p) = \varepsilon$ の時は, $\pi_{1\gamma}(p) = \pi_{2\gamma}(p) = \varepsilon$)

$$\tilde{\pi}_{2\gamma}(p) = \begin{cases} f & \text{if} \quad \pi_{2\gamma}(p) = f\eta \\ \varepsilon & \text{if} \quad \pi_{2\gamma}(p) = \varepsilon \end{cases}$$

$\pi_{2\gamma}(p)$の左端の記号をとりだす関数である。これらの関数において添字γが明らかな時は省略する。

CF文法ではtreeの名称はVの元だけであったので，名称の等しい２つの頂点p_0, p_1を捜すことでuvwxy定理(Bar-Hillel et al [1961])は求められた。しかしIGに対しては，名称にindex部($\pi_2(p)$)が存在するので事情はかなり複雑である。ただ$\vdash_G$の定義からも解るようにIndex部の消費は左端から１個あてであり pushdown記憶機構になっているこの点を考慮して次の定義をする。

[Def 4-2] F^*上の relation $\trianglelefteq$ (cf $\leq$ on J^*)を定義。

$$\eta, \mu \in F^* \text{に対して} \begin{cases} \eta \trianglelefteq \mu & \text{iff } (\exists \xi \in F^*)(\mu = \xi\eta) \\ \eta \triangleleft \mu & \text{iff } \eta \trianglelefteq \mu \wedge \eta \neq \mu \end{cases}$$

[Def 4-3] $\gamma \in \tilde{\mathcal{T}}(G)$に対して，

$$e_\gamma : \mathrm{dom}(\gamma) \longrightarrow 2^{\mathrm{dom}(\gamma)} \qquad \underline{\text{end of parse function}}$$

を次のように定義する。

$$e_\gamma(p) \underset{\mathrm{def}}{=} \left\{ q \in \mathrm{dom}(\gamma) \;\middle|\; \begin{array}{l} p \leq q,\ \pi_2(p) = \pi_2(q), \\ \pi_2(p) \trianglelefteq \pi_2(r) \text{ for } p \leq {}^\forall r \leq q, \\ \pi_2(q \cdot j) \triangleleft \pi_2(p) \text{ for any } j \in J \text{ s.t. } q \cdot j \in \mathrm{dom}(\gamma) \end{array} \right\}$$

直観的には，pのindex部の上に新たなindexが積上げられ，消費されていって，次には$\pi_2(p)$が始めて消費されるその直

前の頂点 q の集合が $e_\gamma(p)$ である。(もちろんそのような q が存在しない時は $e_\gamma(p)=\phi$ である。) P から $e_\gamma(p)$ までを p の scope と呼ぶ。すなわち $scope_\gamma(p)=\{x\in dom(\gamma)\mid p\leq x\wedge(\forall q\in e_\gamma(p))(x\not\geq q)\}$ のことである。

定理の証明のために次の関数は本質的である。

〔Def 4-4〕 $n_\gamma: dom(\gamma)\to 2^N$ は次のように定義される。 $n_\gamma(p)=\{\pi_1(q)\in N\mid q\in e_\gamma(p)\}$

$e_\gamma(p)$ の元の名称の第1成分を集めたものである。

以下 Main Theorem を述べその証明のための基本的 idea を説明し，応用を述べる。

〔定理1〕(生成木増加定理)

" $G=(N,T,F,P,S)$ IG が与えられた時，G にだけ依存する定数 k が存在して次をみたす。

$\#(\hat{\gamma})\geq k$ なる $\gamma\in\mathcal{T}(G)$ に対しては，$\alpha,\beta_i,\delta_i,\tau_i,\nu\in\tilde{\mathcal{T}}(G)$ $(i\geq1)$ が γ より構成的に求めることができて，$\gamma=\alpha\cdot\beta_1\cdot\delta_1\cdot\tau_1\cdot\nu$ と分解され，

i) $\gamma_n=\alpha\cdot\beta_1\cdot\beta_2\cdots\beta_n\cdot\delta_n\cdot\tau_n\cdots\tau_1\cdot\nu\in\mathcal{T}(G)$ $(n\geq1)$
$(\gamma=\gamma_1)$

ii) $\#(\hat{\gamma}_n)<\#(\hat{\gamma}_{n+1})<k_\gamma k^{n+1}$ $(n\geq1)$

ここで k_γ は γ に依存する定数 "

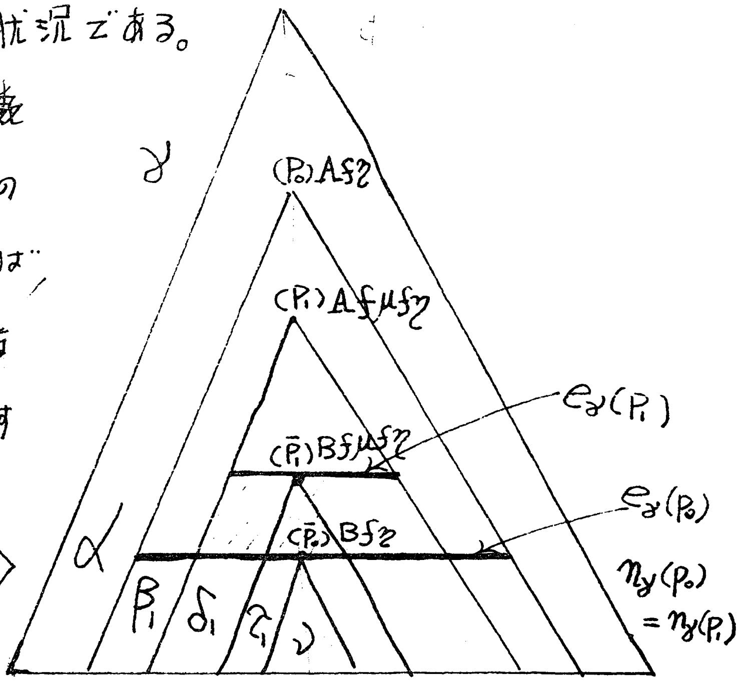
$\alpha,\beta_i,\delta_i,\tau_i,\nu$ の具体的構造は複雑であるので省略する。

【定理1】"証明のための基本的 idea"

(i)の分解は図6のような状況である。すなわち、頂点Pの子の数(枝分れの数)が2以上の時Pを<u>productive</u>と呼べば、γの鎖の中でproductiveな頂点が最も多いものGとする。

$G=\langle (q_0, X_0\eta_0), \ldots, (q_m, X_m\eta_m)\rangle$

$D_G=\{q\in dom(\gamma) \mid (q,\gamma(q))\in G\} - \{q_m\}$ とした時



($\gamma=\gamma_1$の分解図)[図6]

$$S : D_G \longrightarrow N\times\{F\cup\{\varepsilon\}\}\times\{N\cup\{\#\}\}\times 2^N$$

$$S(q)=\begin{cases}(\pi_1(q), \tilde{\pi}_2(q), \pi_1(\bar{e}(q)), \eta_\gamma(q)) & \text{if } \bar{e}(q)\neq\phi \\ (\pi_1(q), \tilde{\pi}_2(q), \#, \eta_\gamma(q)) & \text{if } \bar{e}(q)=\phi\end{cases}$$

ここで#は新しい記号、$\bar{e}(q)=e_\gamma(q)\cap D_G$ を考える。

つまり、$q\in D_G$ に対して、その名称の非終端記号、index部の左端の記号、Gでのqのend of scope、$\bar{e}(q)$の非終端記号、$e(q)$の非終端記号を集めた集合を対応させているのである。

S-functionの値は有限個だから、$S(P_0)=S(P_1)$で

$scope_\gamma(P_1) \subseteq scope_\gamma(P_0)$ なるP_0, P_1の存在を先端の増加に注意をはらって証明すればよい。存在を保証し、か

つ(i),(ii)を満たすようにkを決定できる。$k_\gamma = \#(\hat{\gamma})k^{\#(\hat{\gamma})}$ である。(i)に関しては$S(P_0)=S(P_1)$の条件より、P_0, P_1 間で積み上げられた index $f\mu$ が図6の斜線部で、消費されており、$n_\gamma(P_0)=n_\gamma(P_1)$の条件より $f\mu$ を繰り返し増加させても、うまく消費されてゆくわけである。詳細に関しては〔林〔1972〕)を参照されたい。

定理1の系として次の応用を得る。

§5 応用

【系-1】 Indexed Grammar の finiteness problem は solvable である。(Rounds〔1970〕の別証)

〔証明〕 $G=(N,T,F,P,S)$ IG が与えられた時、ε-free(*) IG $G'=(N',T,F',P',S')$ をGから構成できて、$L(G)=L(G')$ if $\varepsilon \notin L(G)$ or $L(G)=\{\varepsilon\}\cup L(G')$ if $\varepsilon \in L(G)$ とすることが出来る。(Aho〔1968〕, p661)

$L(G)$ finite $\Leftrightarrow$ $L(G')$ finite よってこのG'に対して、定理のkを計算すれば、G'がε-free であることより、$\#(\hat{\gamma})=|g(\gamma)|$ $(\gamma \in J(G'))$ となる。R_kをk以上の長さのTの string から成る正則集合、すなわち $R_k=\{w\in T^* \mid |w|\geq k\}$ とする。すると、$L(G'')=L(G')\cap R_k$ なるIG, $G''=(N'',T,F'',P'',S'')$ が構成できる。(Aho〔1968〕 p656) そして定理1の(ii)より、$L(G')$ finite $\Leftrightarrow L(G'')=\phi$

(*) $A\rightarrow\varepsilon \notin P\cup\bigcup_{f\in F} f$ なるIGをいう。

ところが $L(G'')=\phi$ かどうかは solvable である。(AhO, P658) //

【定理2】 $f:\mathbb{N}\longrightarrow\mathbb{N}$ を $f(n)\leq f(n+1)$ $(n\geq 1)$ かつ, $\overline{\lim_{n\to\infty}} f(n)^{\frac{1}{n}}=\infty$ なる関数とした時,

$L_f=\{a^{f(n)} \mid n\geq 1\}$ は Indexed Language ではない。

[証明] 系-1の時と同様に $L_f-\{\varepsilon\}=L(G)$ なる ε-free IG: G が存在すると仮定して，矛盾を導いても，一般性を失わない。定理1の k をとってくる。$\overline{\lim_{n\to\infty}} f(n)^{\frac{1}{n}}=\infty$ の条件より，$\#(\hat{\gamma})=|g(\gamma)|=f(n_0)\geq k$ なる，n_0，γ が存在する。このγに対して k_γ を計算する。$\overline{\lim_{n\to\infty}} f(n)^{\frac{1}{n}}=\infty$ の条件より，$f(n_0+t)^{\frac{1}{n_0+t}}>k_\gamma k^2$，$(t>1)$ なる t が存在する。定理1のiiより，$f(n_0+t)>(k_\gamma k^2)^{n_0+t}>k_\gamma k^{2t+1}>\#(\hat{\gamma}_{2t+1})=|g(\gamma_{2t+1})|>\#(\hat{\gamma}_{2t})=|g(\gamma_{2t})|>\cdots>|g(\gamma_1)|=f(n_0)$ である。f の単調性より，L_f には長さが $f(n_0)$ より大きく $f(n_0+t)$ より小さい元は高々 $t+1$ 個しか存在しないはずである。ところが上の不等式はそのような元が少くとも $2t+1$ 個存在することを示している。これは矛盾である。 //

定理2を使えば，$\overline{\lim_{n\to\infty}} (n!)^{\frac{1}{n}}=\infty$ であるから $\{a^{n!} \mid n\geq 1\}$ や同様に $\{a^{n^n} \mid n\geq 1\}$ などが Indexed Language でないことが解る。ところが，$P_i\in\mathbb{N}[X]$ を多項式とし，$k_i\in\mathbb{N}$ として，$\varphi(x)=\sum_{i=1}^{j}P_i(x)k_i^{x}$ とおけば，$L_\varphi=\{a^{\varphi(n)} \mid n\geq 1\}$

が Indexed Language であることは容易に解るから，この点

（例えば，$\{a^{nk^n} \mid n \geq 1\}$ を与えるIGは次のとおり。

$G = (\{S, A, B, C\}, \{a\}, \{f, g\}, P, S)$

$P: S \longrightarrow Ag, \quad A \longrightarrow Af, \quad A \longrightarrow B$

$f = [B \longrightarrow B^k C^k, \quad C \longrightarrow C^k], \quad g = [B \rightarrow a^k, C \rightarrow a^k]$

一般に L_φ を与える文法も容易に構成できる。）

を考慮すれば，定理2はIGの限界を与えていると言える。すなわち定理1の ii) から解るように，IGの string の長さの増えかたは高々等比数列的である。（CF文法では等差数列的！）

【定理3】 $L_\Sigma = \{(\# w)^{|w|} \mid w \in \Sigma^*, \# \notin \Sigma, \#(\Sigma) \geq 1\}$

は Indexed Language ではない。

証明には定理1を証明するためのテクニックを拡張して使う。林［1972］を参照されたい。

この定理は，与えられた $k \in N$ に対して $\{(\# w)^k \mid w \in \Sigma^*, \# \notin \Sigma, \#(\Sigma) \geq 1\}$ なる言語を与えるIGは容易に構成できる点を考慮すれば，興味深い。

【謝辞】

本論文を書くにあたって，色々アドバイスをし，議論，討論をして下さった数理解析研究所 高須 達教授，及び高須

研の皆さん，特に西沢輝泰氏と笠井琢美氏に深謝します。

【参考文献】

1. A.V. Aho, "Indexed Grammars". JACM 15 [1968] PP. 647 – 671
2. Bar-Hillel et al, "On Formal Properties of Simple Phrase Structure Grammars". Z. Phonetik Sprachwiss. Kommunikat 14 [1961] PP. 143 – 172
3. W. S. Brainerd, "Tree Generating Regular Systems". Inform & Control 14 [1969] PP. 217 – 231
4. W. F. Ogden, "Intercalation theorems for Pushdown store and Stack languages." Ph. D dissertation. Stanford Univ [1968]
5. W. C. Rounds, "Tree-Oriented Proofs of Some Theorems in Context-free and Indexed Languages". 2nd ACM Symp. Theory on Computing [1970] PP. 109 — 116
6. M. Takahashi, "Derivation Trees and Context free Languages." Masters' thesis Pennsylvania Univ [1970]
7. 林 健志, "Indexed Grammar の木構造について" 京都大学修士論文 [1972]